synthprivacy

Release 1.0.0

EHIL

Jan 26, 2024

CONTENTS

CHAPTER

ONE

ABOUT MEMBERSHIP DISCLOSURE PRIVACY

The *synthprivacy* package currently calculates membership disclosure risk that is associated with synthetic data. It is developed by Electronic Health Information Lab as an implementation of the paper:

El Emam K, Mosquera L, Fang X. Validating a membership disclosure metric for synthetic health data. JAMIA Open. 2022 Oct 11;5(4):ooac083. doi: 10.1093/jamiaopen/ooac083. PMID: 36238080; PMCID: PMC9553223.

CHAPTER

TWO

FUNCTIONS

class synthprivacy.mmbrshp_rsk.**MmbrshpRsk**(*real_data: DataFrame*, *population_size: int*, *h=10*, *seed=13*, *max_no_cores=None*)

Calculates the membership privacy risk associated with synthetic data.

Parameters

- **real_data** (*dataframe*) – Real Dataset as Pandas dataframe.
- **population_size** (*int*) – The population size of from which the real dataset was sampled.
- **h** (*int*) – An integer representing the hamming distance threshold to be used when calculating the membership disclosure risk.
- **seed** (*int*) – An integer for fixing the seed to ensure reproducibility. Set to None if randomness is desired.
- **max_no_cores** (*str*) – Specifies the number of cores as either 'hi' for maximum number of cores, 'lo' for 5 cores or None for no multiprocessing.

quasiID

A list of the names of quais variables to be used in the calculation. If None, all variables will be used.

no_bins

The number of bins to discretize continuous and high cardinality categorical variables.

calc_risk(*syn_data: DataFrame*) → tuple

Calculates the membership disclosure risk.

Parameters

syn_df – Synthetic dataset generated by any model that is trained on the training dataset provided by this class.

Returns

Training and attack data frames. Both dataframes includes an additional column 'ID' for record index track against the real data.

CHAPTER

THREE

EXAMPLES

An example is given in `main.py`. The example uses a dataset imported from the `scikit-learn` library and a generator from `synthcity`. Clearly, you can replace these with your dataset and generative models respectively.

You first instantiate an object using the class `MmbrshpRsk`. The class will partition the input real dataset (using the default parameters) and return a training dataset. Internally, the indices of the training observations are retained for further calculations. You use the training data with any generative model to generate your synthetic data. Finally, you pass the synthetic data to the previously defined `MmbrshpRsk` object to calculate the F1 risk scores. Some parameters can be adjusted for risk calculations, e.g. the hamming distance threshold `h`. If you like to change these parameters, please make sure to change them in the class itself. For further information, please refer to the comments in the script `src/synthprivacy/mmbrshp_rsk.py`.

INDEX